AIAA JOURNAL
Vol. 34, No. 8, August 1996

One-Dimensional Finite Elements Based on the Daubechies Family of Wavelets

Richard D. Patton* and Patrick C. Marks†
Mississippi State University, Mississippi State, Mississippi 39762-5925

The development of a one-dimensional finite element is traced utilizing a Daubechies scaling function as its interpolation function. The element was developed to reduce the computational time and number of degrees of freedom needed to solve vibration and wave propagation problems. This element is used in a dynamic test case, and the results are compared with the results using a standard, constant strain, rod element. The results show a 5:1 reduction in required number of degrees of freedom and in the computational time needed for equivalent accuracy.

Introduction

CONVENTIONAL finite elements are formed using the Rayleigh–Ritz method wherein polynomials are used as approximation functions. In most applications, constant strain or polynomial strain elements are used.[1] To model a complex subject, additional elements are combined with the first, resulting in an approximation of the required solution. This paper discusses a different type of finite element, the wavelet finite element, in which a wavelet approximation function is used.

The wavelet finite element was developed specifically for vibration problems and wave propagation analysis. The concept is that the frequency response of the wavelet's associated scaling function models displacements from 0 Hz to a certain cutoff frequency and thus models vibrations more accurately and with less CPU time than conventional elements.

The finite element model of a vibrating object requires a certain density of elements to accurately simulate vibratory response. This element density is usually given in terms of the wavelength of the highest frequency. The goal of this study is to derive a finite element that will attain a high level of accuracy for a low mesh density as compared with conventional finite elements.

Wavelets were used to solve the one-dimensional wave equation by Bacry et al.[2] Wavelets are used in solving a two-point boundary problem by Beylkin.[3] Sarkar et al. present an investigation of the application of wavelet-like triangular functions to finite elements.[4] The interpolation functions exhibit the translation and dilation properties of wavelets but are not wavelets.

Finite elements were derived in Kurdila et al. based on wavelets generated using affine, fractal interpolation functions and were applied to two-dimensional elasticity problems.[5] A wavelet-based method for solving one-dimensional displacement problems was introduced in Bertoluzza et al.[6]

Wavelet Background

A family of wavelets is composed of the scaling function and an infinite number of wavelets. Because the frequency response of the scaling function matches the frequency profile needed for vibration analysis, this paper concentrates on finite elements using scaling functions.

Scaling functions have two properties that make them valuable: translation and orthonormality.

Beginning with a base scaling function, scaling functions form an orthonormal basis by creating translated scaling functions. These are created by translating the scaling function by an integer.

If the base scaling function is denoted by $\phi(x)$, and has support from [0,17] (i.e., is zero outside of the interval [0,17]), then a first translation would be $\phi(x - 1)$ and would have support on the interval [1,18]. Translations take the general form of

$$\phi(x-n), \qquad n = \text{integer} \tag{1}$$

All scaling functions that are translated by an integer are orthonormal to each other and to the base scaling function.

Scaling functions can have either a compact support or an infinite support in the space domain. Infinite support scaling functions must be truncated. In preliminary work for this paper, the truncation error caused unacceptable numerical errors. All compactly supported scaling functions are called Daubechies scaling functions after their discoverer, I. Daubechies. The process of obtaining these scaling functions is somewhat complex. For details on the formation of scaling functions, the reader is referred to Daubechies.[7]

The formula for the length of the scaling function is $2n - 1$, where n is the order of the wavelet. Short scaling functions are continuous but not differentiable. The relatively long scaling function used in this paper, support of [0,17], was chosen because it is twice differentiable. It is ninth order and therefore is designated the 9ϕ. The 9ϕ scaling function is shown in Fig. 1. Daubechies gives additional scaling functions and wavelets, as well as the algorithm for generating them.[8]

Application of Scaling Functions to Finite Elements

To formulate finite elements based on the Daubechies scaling function, the Ritz method of variational approximation was used.[9] The Ritz method provides an approximate solution to the weak form of wave equation

$$B[v, u(x,t)] = \int_0^L \left(AE \frac{\mathrm{d}v}{\mathrm{d}x} \frac{\mathrm{d}u(x,t)}{\mathrm{d}x} + \rho v \frac{\mathrm{d}^2}{\mathrm{d}t} u(x,t) \right) \mathrm{d}x$$
$$= \int_0^L v w(x,t)\, \mathrm{d}t = L(t) \tag{2}$$

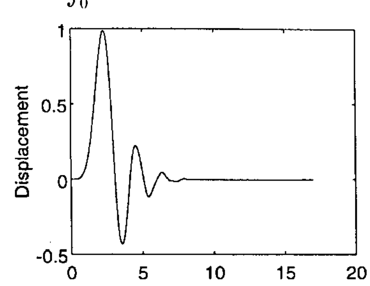

Fig. 1 Daubechies 9ϕ scaling function.

Received Feb. 8, 1995; presented as Papers 95-1444 and 95-1393 at the AIAA/ASME/ASCE/AHS/ASC 36th Structures, Structural Dynamics, and Materials Conference, New Orleans, LA, April 10–12, 1995; revision received Oct. 9, 1995; accepted for publication Oct. 16, 1995.

*Assistant Professor, Department of Mechanical Engineering, Raspet Flight Research Laboratory, P.O. Drawer ME. Member AIAA.

†Research Assistant, Department of Mechanical Engineering, Raspet Flight Research Laboratory, P.O. Drawer ME; currently Development Engineer, Digisonix Inc., 8401 Murphy Drive, Middleton, WI 53562-2543.

where ρ is the mass density of the element, $w(x,t)$ is the applied load, and the displacement, $u(x,t)$, is approximated by

$$u(x,t) = \sum_{i=1}^{N} C_i(t)\xi_i(x) \tag{3}$$

For proper selection of an approximation function, $\xi_i(x)$, the following conditions are usually observed:

1) ξ_i should be sufficiently differentiable, as required by the bilinear form of $B(\xi_i, \xi_j)$.

2) ξ_i should satisfy the boundary conditions of the problem.

3) For any N, the set $\{\xi_i\}_{i=1}^N$ along with the columns (and rows) of $B(\xi_i, \xi_j)$ is linearly independent.

4) $\{\xi_i\}$ is complete.

Scaling functions satisfy all but the second condition. Scaling functions span more than one element. Because of this, the scaling function coefficients satisfy the conditions of continuity between elements. The boundary conditions are applied using a restraint matrix that imposes the boundary conditions on the solution to the problem.

As stated earlier, a scaling function has a characteristic cutoff frequency, ω_0. To illustrate, consider a rod 1 m long, divided into 100 elements (101 grid points). The length between grid points is 0.01 m. If a base scaling function of support [0, 17] were fitted to this grid, it would begin at the zeroth grid point and end at the 17th grid point, covering an interval [0, 0.17]. The rest of the grid can be filled by translating the base scaling function. The first translation would cover the interval [0.1, 0.18]. For a scaling function, the wavelength, λ_0, at the cutoff frequency is always 2 points/wavelength. In this example,

$$\lambda = 2 \times 0.01 = 0.02 \tag{4}$$

$$\omega_0 = 2\pi c/\lambda = 2\pi/0.02 \times c = 100\pi \times c \tag{5}$$

where c = wave speed.

By knowing the maximum frequency of excitation, the grid spacing can be computed. This is the minimum number of points needed for a crude model; more points are needed for an accurate model. If ω_0 is below the excitation frequency, the model has very little response to the excitation.

The scaling function element stiffness matrix is formed using

$$K_{i,j} = AE \int_0^1 \frac{d\phi}{dx}(x+17-i)\frac{d\phi}{dx}(x+17-j)\,dx \qquad 1 \le i, j \le 17 \tag{6}$$

The above equation results in a 17 × 17 element stiffness matrix. The size of the overall stiffness matrix for a model is dependent on the number of elements in the model and the support width of the scaling function. Therefore

initial matrix size = number of elements + support width − 1

For example, a 100-element model would produce a 116 × 116 stiffness matrix.

The scaling function element mass matrix is formed using

$$M_{i,j} = \rho_0 \int_0^1 \phi(x+17-i)\phi(x+17-j)\,dx \qquad 1 \le i, j \le 17 \tag{7}$$

The mass and stiffness matrices are both banded and symmetric. The mass matrix is close to the identity matrix while the stiffness matrix has a band width of $2l - 1$, where l = support width. In this example, the bandwidth is 33.

The boundary conditions are derived as follows. Seventeen scaling functions impinge on any given point. In particular, at $x = 0$, Eq. (3) yields the following for the displacement u:

$$u(0,t) = \sum_{i=-16}^{0} [C_{0,i}(t)\phi(-i)] \tag{8}$$

At the left boundary, $u(0,t) = 0$; therefore

$$\sum_{i=-16}^{0} [C_{0,i}\phi(-i)] = 0 \tag{9}$$

Solving for $C_{0,i}$ with the highest coefficient, $C_{0,-2}$, yields

$$C_{0,-2} = \frac{-1}{\phi(2)}\left[\sum_{i=-16}^{0} C_{0,i}\phi(-i)\right] \tag{10}$$

The left boundary condition R_{ϕ_1} is then formed by letting

$$R_{\phi_1} = [-1/\phi(2)][\phi(16) \quad \phi(15) \quad \cdots \quad \phi(3) \quad \phi(1) \quad \phi(0) \quad 0 \quad \cdots \quad 0] \tag{11}$$

For example, for a 100-element model, let

$$C_0 = [C_{0,-16} \quad C_{0,-15} \quad \cdots \quad C_{0,-3} \quad C_{0,-1} \quad \cdots \quad C_{0,99}]^T \tag{12}$$

Then

$$C_{0,-2} = R_{\phi_1} C_0 \tag{13}$$

At the right boundary, $x = 100$ and $u(100, t) = 0$. Therefore, by similar process,

$$C_{0,98} = R_{\phi_2} C_0 \tag{14}$$

R_{ϕ_1} and R_{ϕ_2} are inserted as extra rows in a 114 × 114 identity matrix to yield R, a 116 × 114 restraint matrix.

The C_0 vector of scaling coefficients can be converted into either strains or displacements directly. To convert the coefficients to displacements, the scaling function is used:

$$u(x,t) = \sum_{i=-16}^{99} C_{0,i}(t)\phi(x-i) \tag{15}$$

where u is the scaling function solution for displacement.

To convert the coefficients to strains, the derivatives are used:

$$\epsilon(x,t) = \sum_{i=-16}^{99} C_{0,i}(t)\frac{d\phi}{dx}(x-i) \tag{16}$$

Using the same 100 element example, the ith element of the load vector $L(t)$, $L_i(t)$, is given by the following:

$$L_i(t) = \int_0^{100} w(x,t)\phi(x-i)\,dx, \qquad -16 \le i \le 99 \tag{17}$$

where $w(x,t)$ is the distributed load on the rod. For this example, consider a sinusoidal load applied at $x = 20$. Therefore $w = \delta(20-x)\sin(wt)$. Then

$$L_i(t) = \int_0^{100} \delta(20-x)\sin(wt)\phi(x-i)\,dx \tag{18}$$

$$L_i(t) = \phi(20-i)\sin(wt) \tag{19}$$

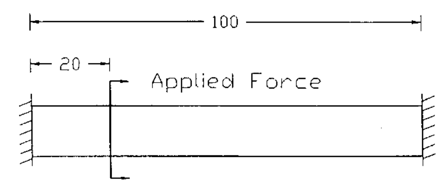

Fig. 2 Schematic of the rod model used in calculations.

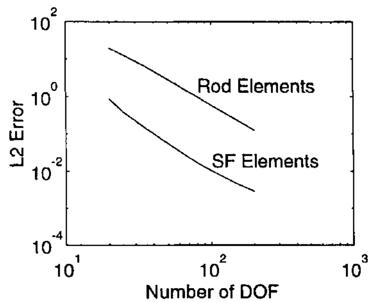

Fig. 3 **L_2 norm of the error vs number of DOF.**

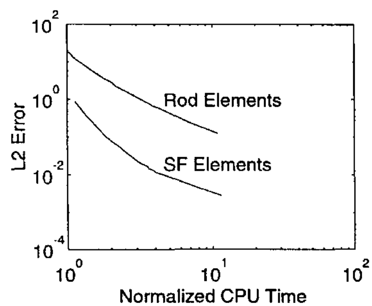

Fig. 4 **L_2 norm of the error vs normalized CPU time.**

Results

Figure 2 shows the model used in the calculations. The rod has a length of 100 units. The model is 100 units long with an excitation force exerted 20 units from the left boundary. It is assumed that $E = A = \rho = 1$. Thus, $c = 1$. The model was excited at a frequency of $\pi/10$, yielding a wavelength of 20.

Using the L_2 norm of the error, the accuracy and computational speed of a model composed of scaling function elements and one composed of lumped mass rod elements were compared. Figure 3 shows a plot of the L_2 norm of the error at time $t = 50$ vs the number of degrees of freedom (DOF) for both models. The scaling function-based elements offer approximately a 5 : 1 decrease in number of DOF for the same accuracy.

In Fig. 4, the relationship between the L_2 norm of the error at time $t = 50$ and the normalized CPU time is shown. The scaling function elements show a substantial decrease (approximately 5:1) in CPU time for a similar error.

Conclusions

A one-dimensional wavelet finite element based on the Daubechies 9ϕ scaling function was developed. The resulting element yields an approximately 5:1 reduction in the number of DOF required to model a vibrating object compared with rod elements. Additionally, for comparable error, the scaling function-based element yields approximately a 5:1 reduction in CPU time vs the rod element.

Acknowledgment

Support for this work was provided by the National Science Foundation Mississippi EPSCoR Advanced Development Program (OSR-9108767).

References

[1]Cook, R. D., Malkus, D. S., and Plesha, M., *Concepts and Applications of Finite Element Analysis*, 3rd ed., Wiley, New York, 1989, pp. 88–100.

[2]Bacry, E., Millat, S., and Papanicolaou, G., "A Wavelet Based Space-Time Adaptive Numerical Method for Partial Differential Equations," *Mathematical Modeling and Numerical Analysis*, Vol. 26, No. 7, 1992, pp. 793–834.

[3]Beylkin, G., "On Wavelet Based Algorithms for Solving Differential Equations," *Wavelets: Mathematics and Applications*, CRC Press, Boca Raton, FL, 1994, pp. 449–466.

[4]Sarkar, T. K., Adve, R., Garcia-Castillo, L. E., and Salazar-Palma, M., "Utilization of Wavelet Concepts in Finite Elements for an Efficient Solution of Maxwell's Equations," *Radio Science*, Vol. 29, No. 4, 1994, pp. 965–977.

[5]Kurdila, A. J., Sun, T., and Gramn, P., "Affine, Fractal Interpolation Functions and Wavelet Based Finite Elements," *Proceedings of the AIAA/ASME/ASCE/AHS/ASC 36th Structures, Structural Dynamics, and Materials Conference and AIAA/ASME Adaptive Structures Forum* (New Orleans), AIAA, Washington, DC, 1995, pp. 2095–2105.

[6]Bertoluzza, S., Naldi, G., and Ravel, J. C., "Wavelet Methods for the Numerical Solution of Boundary Value Problems on the Interval," *Wavelets: Theory, Algorithms, and Applications*, Vol. 5, Academic, San Diego, CA, 1994, pp. 425–448.

[7]Daubechies, I., *Ten Lectures on Wavelets*, Society for Industrial and Applied Mathematics, Philadelphia, 1992, pp. 167–194.

[8]Daubechies, I., *Ten Lectures on Wavelets*, Society for Industrial and Applied Mathematics, Philadelphia, 1992, pp. 195, 205, 206.

[9]Reddy, J. N., *An Introduction to the Finite Element Method*, 2nd ed., McGraw–Hill, New York, 1993, pp. 18–59.